Using Storage Center with FCoE and Linux

Dell Compellent Storage Center

Daniel Tan, Product Specialist
Dell Compellent Technical Solutions

November 2013

Revisions

Date	Revision	Description	Author
Nov 2013	1.0	Initial release	Daniel Tan

© 2013 Dell Inc. All Rights Reserved. Dell, the Dell logo, and other Dell names and marks are trademarks of Dell Inc. in the US and worldwide. All other trademarks mentioned herein are the property of their respective owners.

Table of contents

Revisions 2

Executive summary 4

Purpose 4

1 Leveraging Converged Network Adapters (CNAs) in a converged environment with Linux 5

1.1 FCoE based Block Storage 5

1.2 iSCSI based Block Storage 15

1.3 Special considerations 22

2 Leveraging traditional 10Gb Ethernet NICs in a converged environment with Linux 24

2.1 Performance considerations 29

2.2 Special considerations 29

A Additional resources 30

Executive summary

Managing and scaling network infrastructure is often an expensive proposition for organizations of any size. It is for the above reasons and more, whereby system managers' and system administrators' try to leverage and optimize the use of their existing infrastructure without compromising their network needs or performance. This paper attempts to address some of the above concerns by leveraging the use of Linux with the open source open-FCoE IO stack to achieve competitive SAN delivery and performance in use with Dell Compellent Storage Center over any existing Ethernet infrastructure.

Purpose

This paper outlines and defines the administrative concepts and tasks regarding setting up, configuring and using Linux with Fibre Channel over Ethernet (FCoE) with Dell Compellent Storage Center. As is common with Unix/Linux environments, there are various ways to accomplish any particular task; the examples in this paper merely represent one of these various ways to accomplish these tasks.

1 Leveraging Converged Network Adapters (CNAs) in a converged environment with Linux

1.1 FCoE based Block Storage

This section outlines the installation and configuration of a Linux server leveraging hardware-based Converged Network Adapters (CNAs) in a converged connectivity environment. In this environment, the physical server has a two-port CNA card installed, with each port connected to a 10Gb Fiber Chanel over Ethernet (FCoE) capable switch and will provide the server with both Fibre Channel and iSCSI based block storage, as well as high performance traditional Ethernet connectivity.

The configuration of the CNAs in the BIOS menus will appear similar to that of traditional Fibre Channel HBAs. It would be recommended at this time to note and record the respective MAC addresses of each CNA, which may be very useful when needing to troubleshoot configurations from the network perspective.

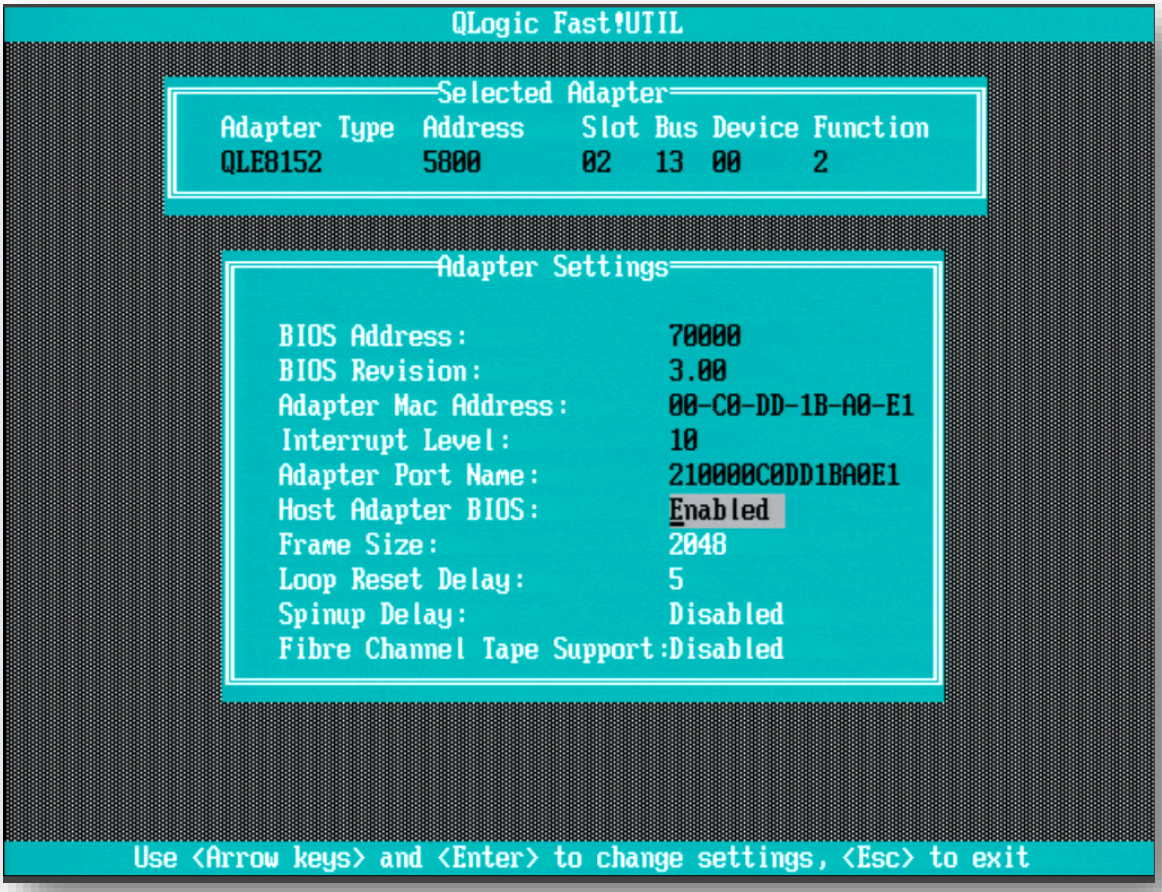

Figure 1 CNA BIOS configuration

In the advanced configuration options, notice the "Link Down Timeout" setting; configuring this to 60 seconds in the BIOS is an important and required step for surviving Storage Center failures or controller failover events. Additionally, observe the addition of the "Primary FCF VLAN ID" setting, which may be required for managing VLANs in some first generation FCoE/CNA environments.

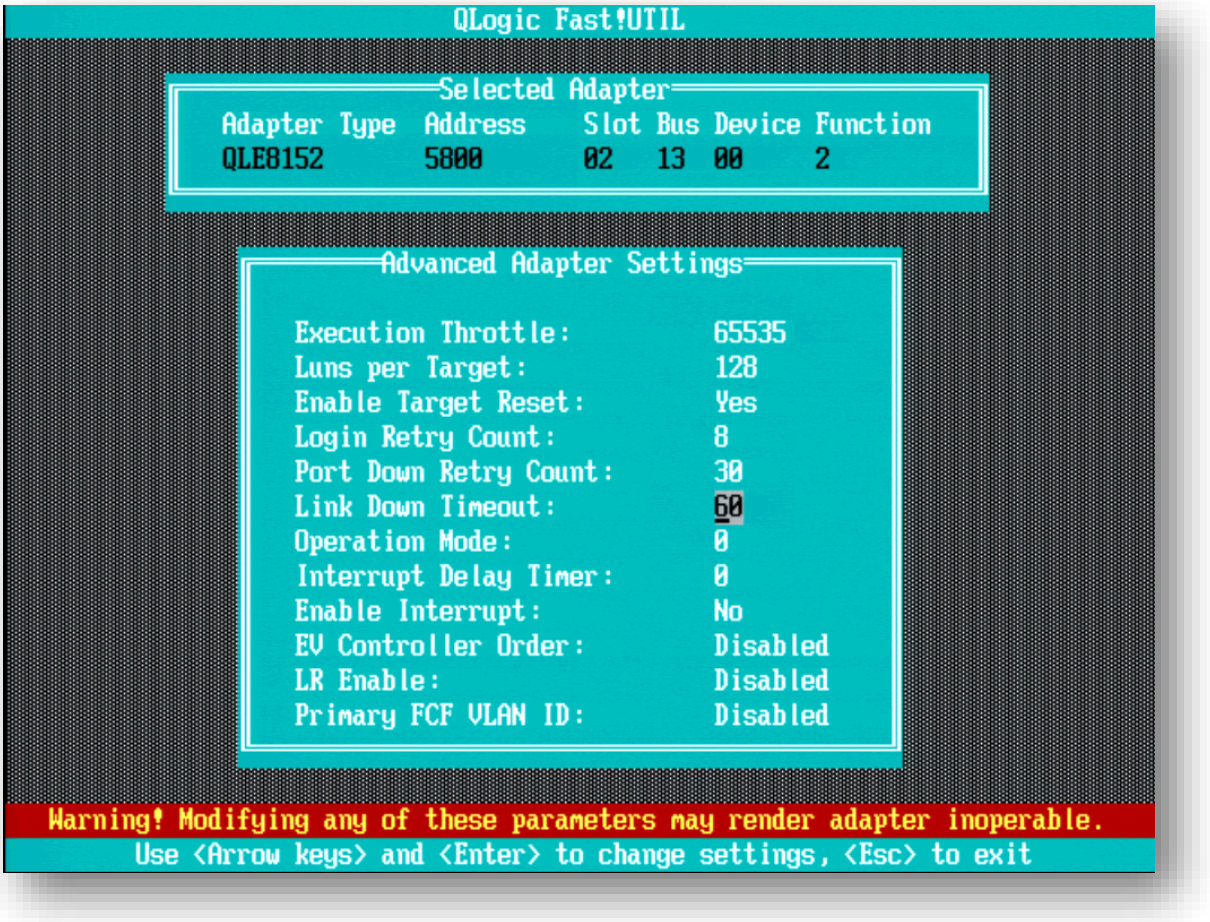
QLogic Fast!UTIL
Selected Adapter
Adapter Type Address Slot Bus Device Function
QLE8152 5800 02 13 00 2
Advanced Adapter Settings
Execution Throttle: 65535
Luns per Target: 128
Enable Target Reset: Yes
Login Retry Count: 8
Port Down Retry Count: 30
Link Down Timeout: 60
Operation Mode: 0
Interrupt Delay Timer: 0
Enable Interrupt: No
EV Controller Order: Disabled
LR Enable: Disabled
Primary FCF VLAN ID: Disabled
Warning! Modifying any of these parameters may render adapter inoperable.
Use <Arrow keys> and <Enter> to change settings, <Esc> to exit

Figure 2 "Link Down Timeout" configuration

Once the appropriate zoning and connectivity configurations have been configured, the CNAs should have visibility into the Storage Center SAN as shown below. The CNAs can be rescanned and a new Server Object created within the Storage Center interface.

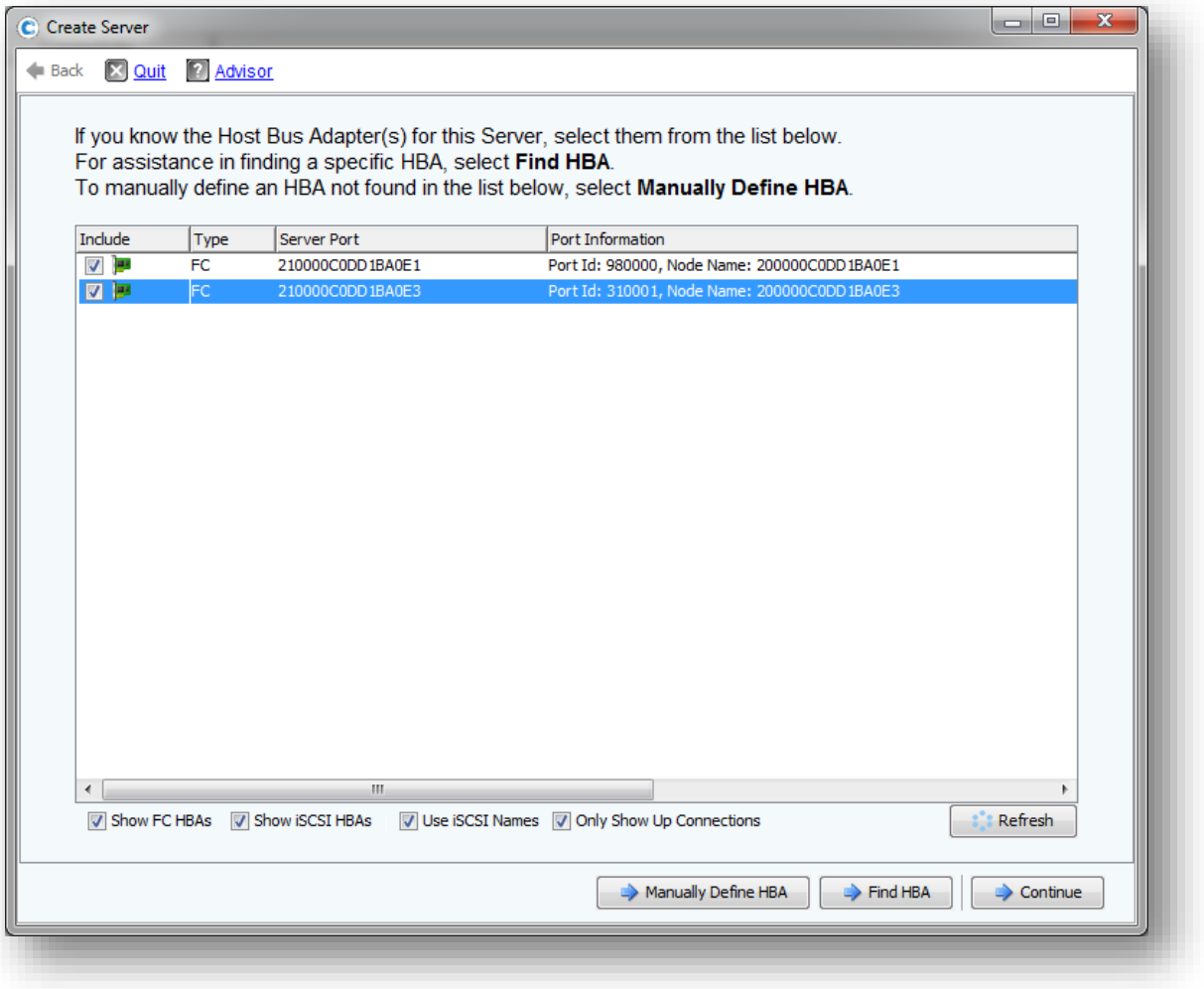

Figure 3 Setting up new Server Object

After the Server Object has been created in the Storage Center, a boot volume can be created and mapped (as LUN ID 0) to this Server Object. The CNAs can then be configured to use the volume as it's default boot target as shown below.

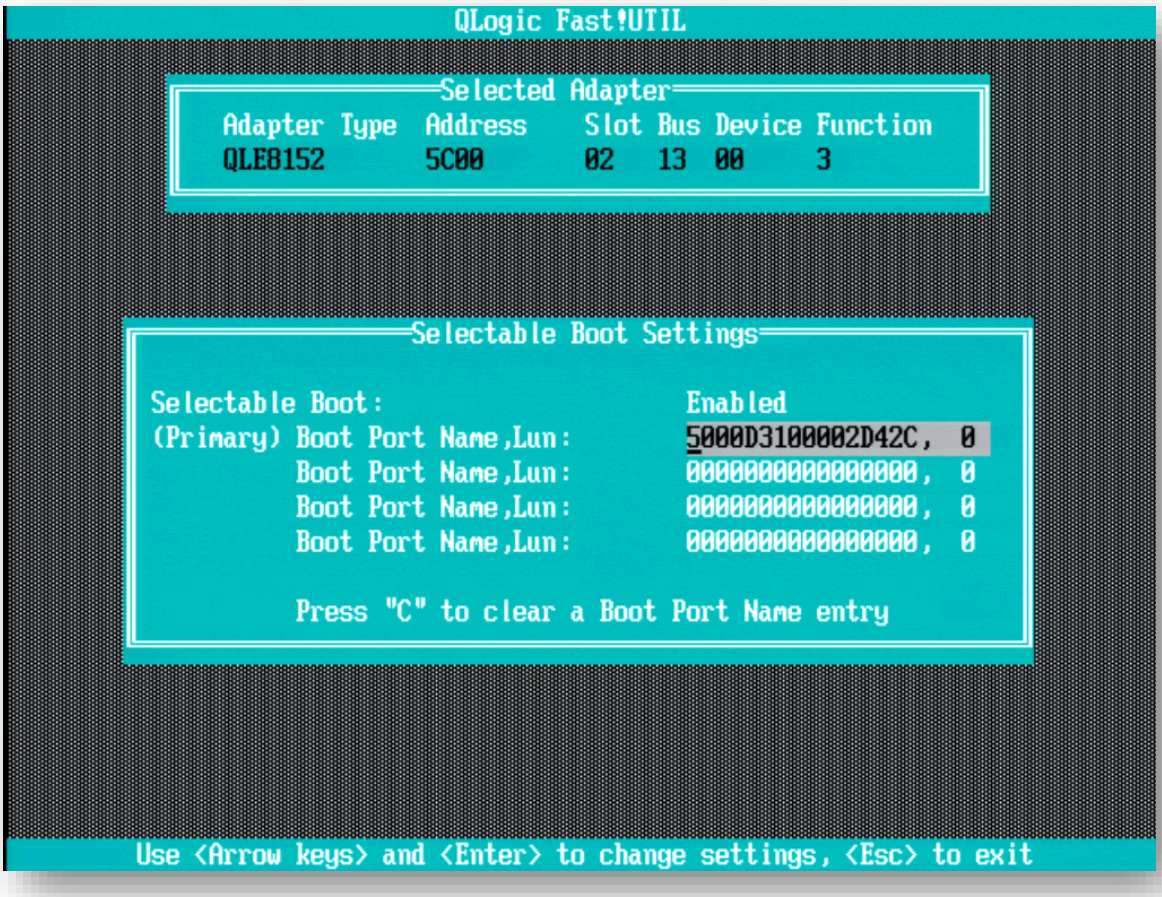
QLogic Fast!UTIL
Selected Adapter
Adapter Type Address Slot Bus Device Function
QLE8152 5C00 02 13 00 3
Selectable Boot Settings
Selectable Boot: Enabled
(Primary) Boot Port Name,Lun: 5000D3100002D42C, 0
Boot Port Name,Lun: 0000000000000000, 0
Boot Port Name,Lun: 0000000000000000, 0
Boot Port Name,Lun: 0000000000000000, 0
Press "C" to clear a Boot Port Name entry
Use <Arrow keys> and <Enter> to change settings, <Esc> to exit

Figure 4 Configure boot from SAN target in BIOS

At this point, the Linux install process can proceed. As is reflected in the Redhat Enterprise Linux Best Practices guide, it is recommended to configure multipath boot from SAN during the installation process. Using the Red Hat Enterprise Linux 6 installer as an example, only two configuration steps are required to instruct the installer to configure multipath boot from SAN. The "Specialized Storage" menu exposes the multipath object as an installation target for further partitioning and file system layout as shown below.

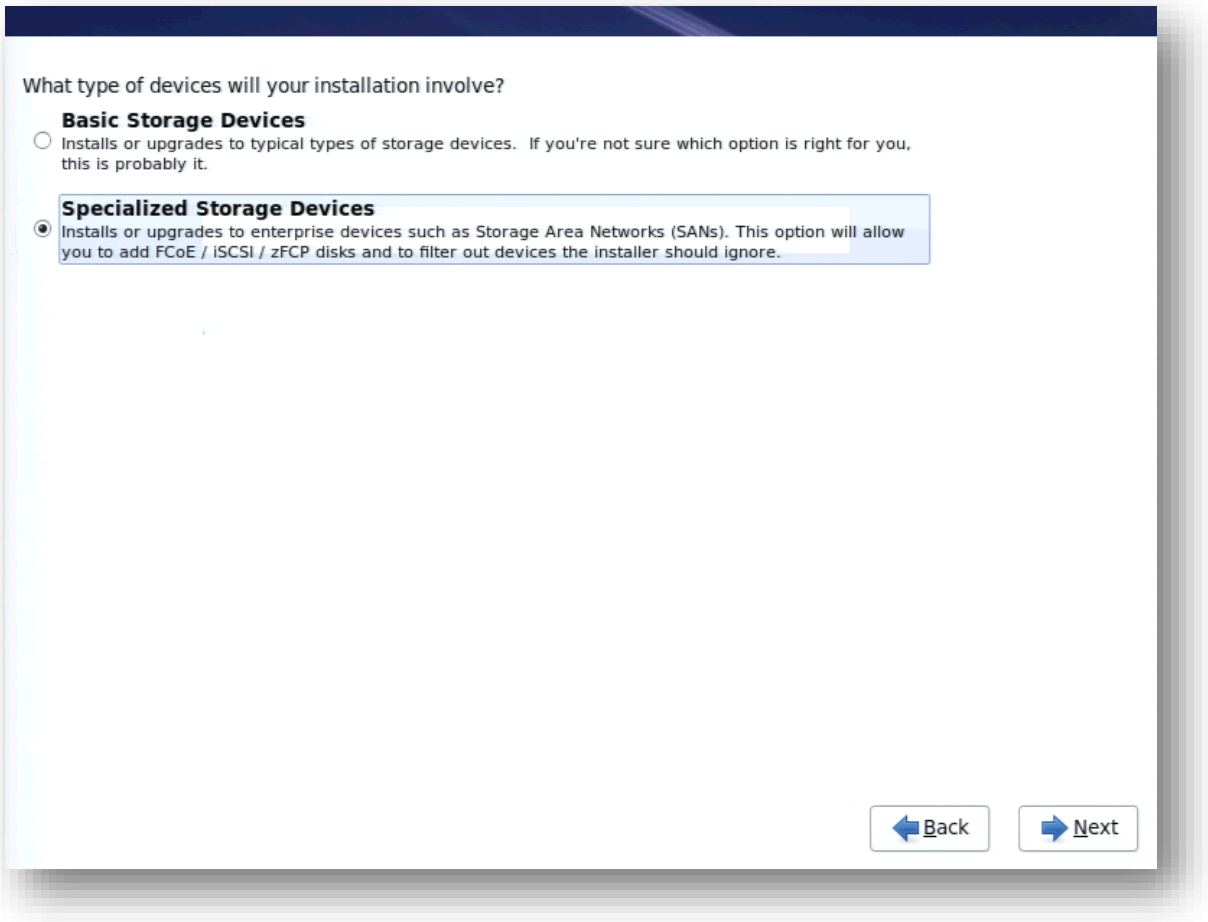
What type of devices will your installation involve?
Basic Storage Devices
Installs or upgrades to typical types of storage devices. If you're not sure which option is right for you, this is probably it.
Specialized Storage Devices
Installs or upgrades to enterprise devices such as Storage Area Networks (SANs). This option will allow you to add FCoE / iSCSI / zFCP disks and to filter out devices the installer should ignore.
Back
Next

Figure 5 Select "Specialized Storage Devices" during Redhat installation

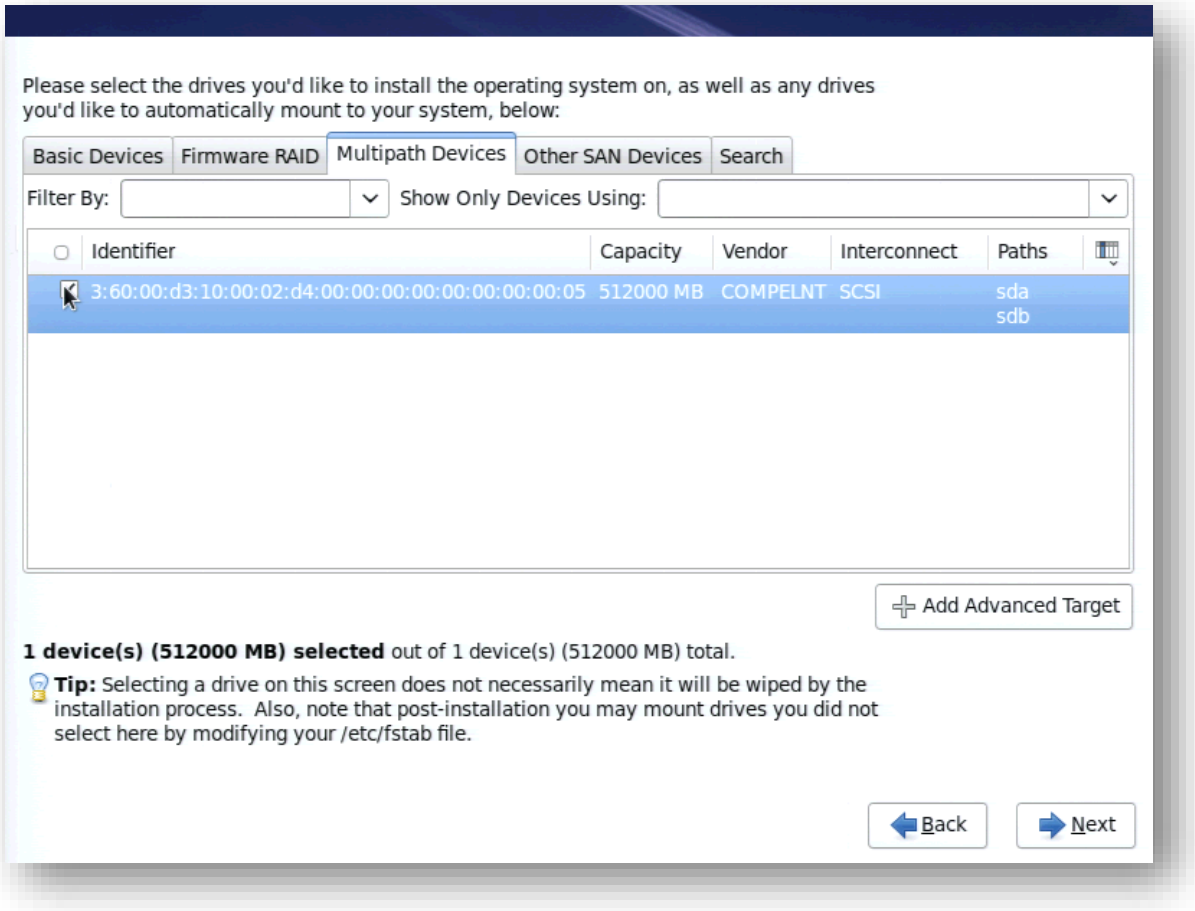
Please select the drives you'd like to install the operating system on, as well as any drives you'd like to automatically mount to your system, below:
Basic Devices
Firmware RAID
Multipath Devices
Other SAN Devices
Search
Filter By:
Show Only Devices Using:
Identifier
Capacity
Vendor
Interconnect
Paths
3:60:00:d3:10:00:02:d4:00:00:00:00:00:00:00:00:05
512000 MB
COMPELNT
SCSI
sda
sdb
Add Advanced Target
1 device(s) (512000 MB) selected out of 1 device(s) (512000 MB) total.
Tip: Selecting a drive on this screen does not necessarily mean it will be wiped by the installation process. Also, note that post-installation you may mount drives you did not select here by modifying your /etc/fstab file.
Back
Next

Figure 6 Select the boot from SAN Storage Center volume as the installation target

It's important to note that the "Add Advanced Target/FCoE" menu is not required for this sort of installation. Because the CNAs present the raw FCoE connected volumes as if they were attached local disk, no extra steps are required. After selecting the multipath device, the installer can proceed as it were any other Linux installation. After the installation, the multipath command may be used to confirm the proper installation of the OS to the multipath boot from SAN target.

```
# multipath -ll

mpatha (36000d3100002d4000000000000000005) dm-0 COMPELNT,Compellent Vol

size=500G features='1 queue_if_no_path' hwhandler='0' wp=rw

`-+- policy='round-robin 0' prio=1 status=active

|- 0:0:3:0 sda 8:0 active ready running

`- 1:0:3:0 sdb 8:16 active ready running
```

In line with the Redhat Enterprise Linux Best Practices guide (see Appendix A), the CNA driver should be configured to take advantage of the environment. As seen in the previous multipath output, multipath by default uses the "queue_if_no_path" setting, which will allow the Linux server to survive Storage Center controller failover events. Because "queue_if_no_path" is trusted to allow the server to survive failover events, the CNA driver timeout value can thus be reduced to a lower threshold. This will cause multipath to fail a path quickly in the event that path is lost (pulling a cable), and send outstanding I/O down the remaining path. The default size for the driver I/O queue can also be increased, in order to provide the best throughput out of each individual path. In order to check the current values, the following two files can be reviewed as below.

```
# cat /sys/module/qla2xxx/parameters/qlport_down_retry

0

# cat /sys/module/qla2xxx/parameters/ql2xmaxqdepth

32
```

In Redhat Enterprise Linux 6, changing these values can be done by creating or editing the file "/etc/modprobe.d/qla2xxx.conf"and entering a line similar to "options qla2xxx qlport_down_retry=5 ql2xmaxqdepth=64". After the options have been specified in the configuration file, the ramdisk used to boot Linux from the boot loader (located in /boot) needs to be updated to incorporate the new variables into the environment at boot time. After first backing up the initramfs files in the /boot directory, this can be done using the "dracut" command as below.

```
# dracut –f
```

Once the server has been rebooted, the change in the appropriate settings can be confirmed with the commands below.

```
# cat /sys/module/qla2xxx/parameters/qlport_down_retry

5

# cat /sys/module/qla2xxx/parameters/ql2xmaxqdepth

64
```

There is no one size fits all for every environment. The appropriate values for these settings is environment centric. A server with only one path for example should set the "qlport_down_retry" value to 60 in order to properly survive failover events and environments that are exceptionally sensitive to latency may see a benefit in reducing the "ql2xmaxqdepth" value.

Because the CNA abstracts the majority of FCoE connectivity information from the host, troubleshooting FCoE connectivity can be challenging. The tools provided by the CNA vendors can be useful resources in troubleshooting or identifying issues. In this example, the QLogic "scli" tool may provide valuable information about the CNAs and the networks attached to them.

```
# scli
Scanning QLogic FC HBA(s) and device(s), please wait...
-
SANsurfer FC/CNA HBA CLI
v1.7.3 Build 40
Main Menu
1: General Information
2: HBA Information
3: HBA Parameters
4: Target/LUN List
5: iiDMA Settings
6: Boot Device
7: Utilities
8: Beacon
9: Diagnostics
10: Statistics
11: FCoE
12: Help
13: Exit
Enter Selection: 11
SANsurfer FC/CNA HBA CLI
v1.7.3 Build 40
FCoE Utilities Menu
HBA Model QLE8152
1: Port 1: WWPN: 21-00-00-C0-DD-1B-A0-E1 Online
2: Port 2: WWPN: 21-00-00-C0-DD-1B-A0-E3 Online
```

```
3: Return to Previous Menu

Note: 0 to return to Main Menu

Enter Selection:
```

In the output shown above, the "Online" postfix behind each HBA Port identifier represents that the CNA has negotiated with the network and can be considered to be in a "Link Up" state in the attached FCoE network. In the output below, further HBA details are shown.

```
-------------------------------------------------------------------------------

HBA Instance 0: QLE8152 Port 1 WWPN 21-00-00-C0-DD-1B-A0-E1 PortID 31-00-0A

-------------------------------------------------------------------------------

------------------------------------------------------

General Info

------------------------------------------------------

MPI FW Version : 01.40.03

EDC FW Version : 01.08.00

VN Port MAC Address : 0E:FC:00:31:00:0A

VLAN ID : 300

Max Frame Size : 2500 (Baby Jumbo)

Addressing Mode : FPMA

------------------------------------------------------

Hit <RETURN> to continue:
```

The CNA will additionally identify the VLAN available to it, with most switching vendors, this means the port can be easily correlated to its associated VLAN to fabric mapping. The TLV settings for a CNA will expose any DCB policies inherited from the FCoE network as shown below.

```
TLV Menu

HBA Instance 0 (QLE8152 Port 1) : Online

ENode MAC Addr: 00:C0:DD:1B:A0:E1

WWPN : 21-00-00-C0-DD-1B-A0-E1

Desc : QLE8152 QLogic PCI Express to 10 GbE Dual Channel CNA (FCoE)
```

```
1: Details

2: Raw

3: Return to Previous Menu

Note: 0 to return to Main Menu

Enter Selection: 1

-----------------------------------------------------

DCBX Parameters Details for CNA Instance 0 - QLE8152

-----------------------------------------------------

Sat Jul 16 14:07:37 2011

DCBX TLV (Type-Length-Value) Data

================================

DCBX Parameter Type and Length

DCBX Parameter Length: 13

DCBX Parameter Type: 2

DCBX Parameter Information

Parameter Type: Current

Pad Byte Present: Yes

DCBX Parameter Valid: Yes

Reserved: 0

DCBX Parameter Data
```

1.2 iSCSI based Block Storage

The Ethernet interface functions on these installed CNAs can also be configured and used for general connectivity and hence also support high speed 10Gb iSCSI. This section will discuss the configuration of the 10Gb CNA interfaces and iSCSI in a converged network.

Because the connectivity practices may vary between FCoE connectivity vendors, this example will only discuss configuring both interfaces as VLAN-aware; this will facilitate increased flexibility for various vendors at the networking level.

Both eth0 and eth1 will be configured to each support a VLAN-aware virtual interface. First, the base interface must be configured to best support subsequent virtual interfaces.

```
# cat ifcfg-eth0

DEVICE="eth0"

#HWADDR="00:C0:DD:1B:A0:E0"

NM_CONTROLLED="yes"

ONBOOT="yes"

MTU=9000

BOOTPROTO=none

TYPE=Ethernet

# cat ifcfg-eth1

DEVICE="eth1"

HWADDR="00:C0:DD:1B:A0:E2"

NM_CONTROLLED="yes"

ONBOOT="yes"

MTU=9000

BOOTPROTO=none

TYPE=Ethernet
```

In the scenario below, one interface is receiving VLAN 550 tagged frames, and the other is receiving VLAN 10 tagged frames. Each physical interface must be associated with a virtual interface that is VLAN aware and has an IP from a subnet that is available on the VLAN.

```
# cat ifcfg-eth0.550

DEVICE="eth0.550"

ONBOOT="yes"

BOOTPROTO=static

IPADDR=172.16.26.32

NETMASK=255.255.240.0

MTU=9000

VLAN=yes
```

```
# cat ifcfg-eth1.10

DEVICE="eth1.10"

ONBOOT="yes"

BOOTPROTO=static

IPADDR=10.10.26.95

NETMASK=255.255.0.0

MTU=9000

VLAN=yes
```

Once the interfaces have been enabled, jumbo frames connectivity can be confirmed by sending large packet pings to another jumbo frame enabled host on the network.

```
# ping -M do -s 8972 10.10.26.151

PING 10.10.26.151 (10.10.26.151) 8972(9000) bytes of data.

8980 bytes from 10.10.26.151: icmp_seq=1 ttl=64 time=0.137 ms

8980 bytes from 10.10.26.151: icmp_seq=2 ttl=64 time=0.129 ms
```

Once healthy connectivity has been confirmed, the process of configuring iSCSI can begin. In this example "/etc/iscsi/iscsid.conf" will be edited to change the value of "node.session.queue_depth" from "32" to "64" in an attempt to increase throughput.

The process of configuring the iSCSI targets begins with iSCSI discovering available targets on the Storage Center and logging into those targets addresses. It should be noted that the Storage Center being targeted by iSCSI is an entirely separate Storage Center from the Storage Center providing FCoE connected storage. This highlights the power and flexibility of a converged network. The iSCSI targeted Storage Center is using Virtual Ports with 2 independent fault domains (the two VLANs that the Linux server is aware of). Because it is using Virtual Ports, the control ports for each fault domain can be targeted, and each control port will report back all available targets in its fault domain, simplifying the configuration process.

```
# iscsiadm -m discovery -t st -p 10.10.140.4

10.10.140.4:3260,0 iqn.2002-03.com.compellent:5000d31000006923

10.10.140.4:3260,0 iqn.2002-03.com.compellent:5000d31000006925

# iscsiadm -m discovery -t st -p 172.16.26.4

172.16.26.4:3260,0 iqn.2002-03.com.compellent:5000d31000006922
```

```
172.16.26.4:3260,0 iqn.2002-03.com.compellent:5000d31000006924
```

Now that the targets have been discovered, login requests can be sent, which will make the iSCSI initiator iSCSI iqn a visible object on the Storage Center.

```
# iscsiadm -m node --login

Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d31000006922, portal: 172.16.26.4,3260]

Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d31000006923, portal: 10.10.140.4,3260]

Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d31000006925, portal: 10.10.140.4,3260]

Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d31000006924, portal: 172.16.26.4,3260]

Login to [iface: default, target: iqn.2002-03.com.compellent:5000d31000006922,
portal: 172.16.26.4,3260] successful.

Login to [iface: default, target: iqn.2002-03.com.compellent:5000d31000006923,
portal: 10.10.140.4,3260] successful.

Login to [iface: default, target: iqn.2002-03.com.compellent:5000d31000006925,
portal: 10.10.140.4,3260] successful.

Login to [iface: default, target: iqn.2002-03.com.compellent:5000d31000006924,
portal: 172.16.26.4,3260] successful.
```

From the Storage Center interface, the Server Object can now be created. During the creation, notice the two IPs listed, this means the Storage Centers has observed logins from the listed iSCSI iqn from two unique and distinct subnets, representing a healthy multipath environment.

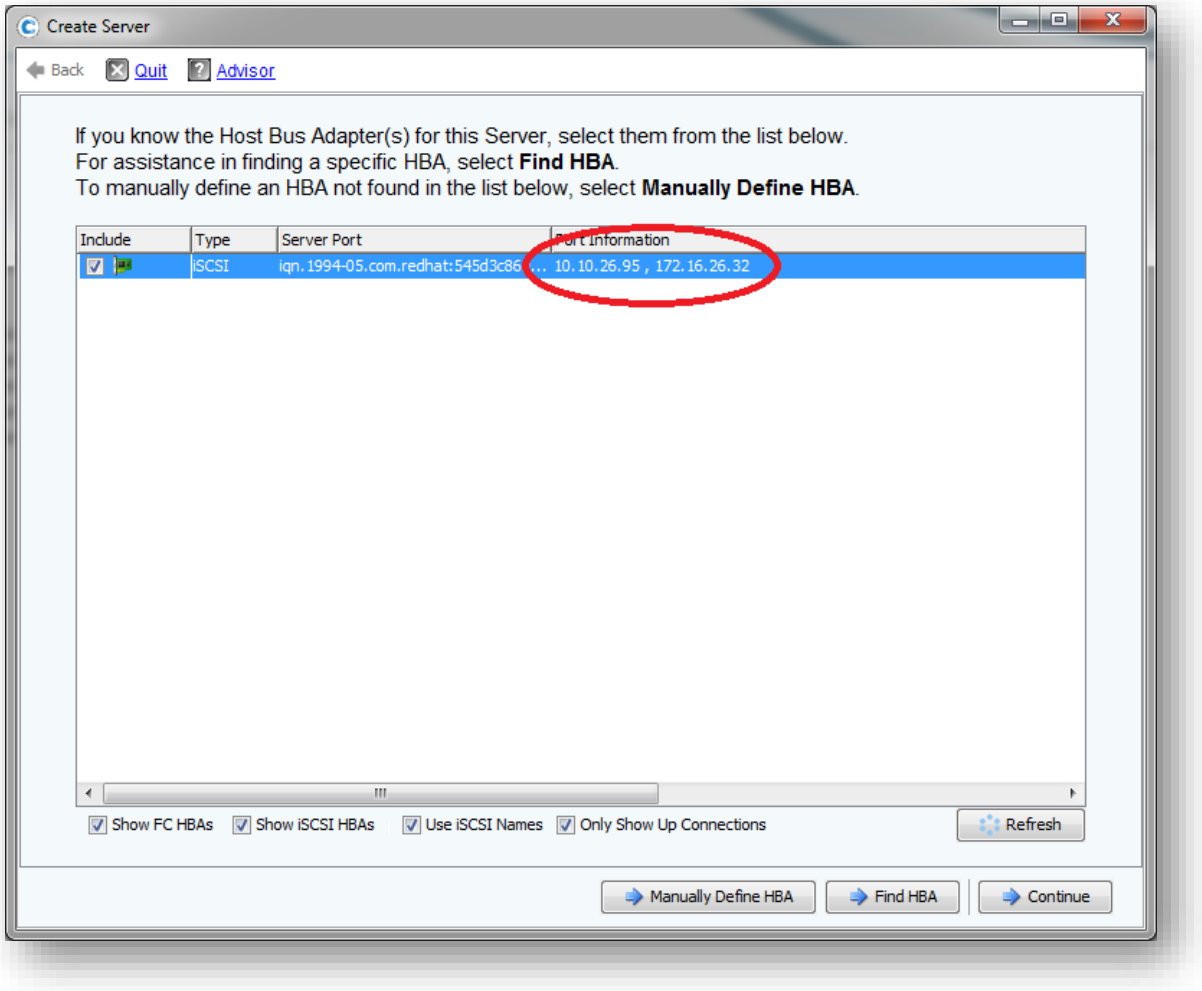

Figure 7 Observe the iqn is represented by two (2) unique IP addresses

Once the Server Object has been created and a volume mapped to it, the iSCSI connections can be rescanned and discovered from the Linux server.

```
# iscsiadm -m node -R

Rescanning session [sid: 1, target: iqn.2002-03.com.compellent:5000d31000006922,
portal: 172.16.26.4,3260]

Rescanning session [sid: 2, target: iqn.2002-03.com.compellent:5000d31000006923,
portal: 10.10.140.4,3260]

Rescanning session [sid: 3, target: iqn.2002-03.com.compellent:5000d31000006925,
portal: 10.10.140.4,3260]
```

```
Rescanning session [sid: 4, target: iqn.2002-03.com.compellent:5000d31000006924,
portal: 172.16.26.4,3260]
```

Looking at the messages log, we can see that new disk devices have been created.

```
# tail /var/log/messages

Jul 11 20:16:23 techtip kernel: sd 6:0:0:1: [sdc] Write Protect is off

Jul 11 20:16:23 techtip kernel: sd 6:0:0:1: [sdc] Write cache: disabled, read
cache: enabled, doesn't support DPO or FUA

Jul 11 20:16:23 techtip kernel: sd 7:0:0:1: [sdd] Write Protect is off

Jul 11 20:16:23 techtip kernel: sd 7:0:0:1: [sdd] Write cache: disabled, read
cache: enabled, doesn't support DPO or FUA

Jul 11 20:16:23 techtip kernel: sdc: unknown partition table

Jul 11 20:16:23 techtip kernel: sd 6:0:0:1: [sdc] Attached SCSI disk

Jul 11 20:16:23 techtip kernel: sdd: unknown partition table

Jul 11 20:16:23 techtip kernel: sd 7:0:0:1: [sdd] Attached SCSI disk

Jul 11 20:16:23 techtip multipathd: sdc: add path (uevent)

Jul 11 20:16:23 techtip multipathd: sdd: add path (uevent)
```

The command scsi_id can be used to discover the unique name for the two paths.

```
# scsi_id -u -g /dev/sdc

36000d31000006900000000000000efb

# scsi_id -u -g /dev/sdd

36000d31000006900000000000000efb
```

Because both devices have the same unique name, they are actually simply two paths to the same actual volume on the Storage Center. In order to create a multipath object across both paths, the /etc/multipath.conf file needs to be edited (note the existence of another multipath object, this is the boot volume from the FCoE connected Storage Center).

```
blacklist_exceptions {

wwid "36000d3100002d4000000000000000005"

wwid "36000d31000006900000000000000efb"

}
```

```
multipaths {
multipath {
uid 0
alias mpatha
gid 0
wwid "36000d3100002d4000000000000000005"
mode 0600
}
multipath {
alias mpathiscsivol01
wwid "36000d3100006900000000000000efb"
}
}
```

Once the multipath daemon service has been restarted, the new volume should appear as a multipath object.

```
# multipath -ll
mpatha (36000d3100002d4000000000000000005) dm-0 COMPELNT,Compellent Vol
size=500G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=1 status=active
|- 0:0:3:0 sda 8:0 active ready running
`- 1:0:3:0 sdb 8:16 active ready running
mpathiscsivol01 (36000d3100006900000000000000efb) dm-6 COMPELNT,Compellent Vol
size=10T features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=1 status=active
|- 6:0:0:1 sdc 8:32 active ready running
`- 7:0:0:1 sdd 8:48 active ready running
```

Instead of using LVM, a file system will simply be laid across the entire volume. For more information about LVM and Storage Center, please consult the Linux Best Practices paper.

```
# mkfs.xfs /dev/mapper/mpathiscsivol01

log stripe unit (2097152 bytes) is too large (maximum is 256KiB)

log stripe unit adjusted to 32KiB

meta-data=/dev/mapper/mpathiscsivol01 isize=256 agcount=33, agsize=83885568 blks

= sectsz=512 attr=2

data = bsize=4096 blocks=2684354560, imaxpct=5

= sunit=512 swidth=512 blks

naming =version 2 bsize=4096 ascii-ci=0

log =internal log bsize=4096 blocks=521728, version=2

= sectsz=512 sunit=8 blks, lazy-count=1

realtime =none extsz=4096 blocks=0, rtextents=0
```

Finally, a mount point and fstab entry has to be created, and the volume mounted. Notice the "_netdev" mount flag that is entered into the fstab file, this ensures that no attempt to mount the file system is made till the network is brought up and the iSCSId daemon has been started at boot time.

```
# mkdir /var/backups

# echo "/dev/mapper/mpathiscsivol01 /var/backups/ xfs _netdev,noatime 0 0" >>
/etc/fstab & mount –a
```

The configuration is now complete, the Linux server is now accessing storage presented by two different Storage Centers using two storage protocols all across one converged logical network.

1.3 Special considerations

By default, if an Fibre Channel HBA and a FCoE CNA from the same vendor are installed on the same physical host, it is likely that they will share a common driver (qla2xxx for example). This may potentially cause problems if the connectivity has different characteristics, both performance and especially topology characteristics. For example, if the Fibre Channel connectivity provides multiple paths, but the FCoE connectivity does not, then the timeout values for the driver will need to be set high, in order to allow the non-multipath managed volumes to survive Storage Center failover events. This sort of dilemma can be overcome in multiple ways. Some examples are to have Linux multipath manage all volumes regardless of the absence of multiple paths, and use the "queue_if_no_path" flag to guarantee surviving Storage Center controller failovers. Another example is by the use of the hardware model specific proprietary drivers,

which will then separate out management and configuration to each individual driver and their respective and unique configuration files.

2 Leveraging traditional 10Gb Ethernet NICs in a converged environment with Linux

In most environments, converged Fibre Channel over Ethernet networks can only be leveraged by using expensive Converged Network Adapters, which abstract the connectivity in front of the host. Through the use of open source tools (open-FCoE), Linux has the unique ability to emulate and speak FCoE inside of the host operating system, allowing Linux to connect to FCoE networks with traditional, less expensive 10Gb Ethernet NICs. An example of where this functionality could be exceptionally useful is in environments that are or will be migrated to an Ethernet only infrastructure. In this sort of environment, the servers can be provisioned with pure Ethernet only connectivity (no expensive CNAs), but still have access to legacy Fibre Channel only accessible storage.

In this example, a Linux server will be connected to a Storage Center volume over a 10Gb Ethernet connection, using the native Linux fcoe-utils on a CentOS-based host.

The fcoe-utils package must first be installed.

```
# yum install fcoe-utils

[snip]

Installed:

fcoe-utils.x86_64 0:1.0.14-9.el6

Dependency Installed:

libconfig.x86_64 0:1.3.2-1.1.el6 libhbaapi.x86_64 0:2.2-10.el6

libhbalinux.x86_64 0:1.0.10-1.el6 libpciaccess.x86_64 0:0.10.9-2.el6

lldpad.x86_64 0:0.9.38-3.el6 vconfig.x86_64 0:1.9-8.1.el6

Complete!
```

In order to leverage multipath, two 10Gb interfaces will be enabled, each requires a configuration file in /etc/fcoe.

```
# cp /etc/fcoe/cfg-ethx /etc/fcoe/cfg-eth4

# cp /etc/fcoe/cfg-ethx /etc/fcoe/cfg-eth5
```

In order to facilitate the network aware settings required for healthy FCoE networks, the lldpad service, which understands and speaks the Link Layer Discovery Protocol (LLDP), must be started, and configured to start on boot.

```
# /etc/init.d/lldpad start

# chkconfig lldpad on
```

The “dcbtool” is used to configure the parameters needed to allow FCoE traffic over the Ethernet links.

```
# dcbtool sc eth4 dcb on

Version: 2

Command: Set Config

Feature: DCB State

Port: eth4

Status: Successful

# dcbtool sc eth4 pfc e:1

Version: 2

Command: Set Config

Feature: Priority Flow Control

Port: eth4

Status: Successful

# dcbtool sc eth4 app:fcoe e:1

Version: 2

Command: Set Config

Feature: Application FCoE

Port: eth4

Status: Successful

# dcbtool sc eth5 dcb on

Version: 2

Command: Set Config

Feature: DCB State

Port: eth5

Status: Successful

# dcbtool sc eth5 pfc e:1

Version: 2
```

```
Command: Set Config

Feature: Priority Flow Control

Port: eth5

Status: Successful

# dcbtool sc eth5 app:fcoe e:1

Version: 2

Command: Set Config

Feature: Application FCoE

Port: eth5

Status: Successful
```

In order for FCoE connectivity to be established, Linux must be configured to enable the interfaces, otherwise no traffic will be able to pass. For ease of use, configuring the interfaces to establish link at boot time is suggested. The below "/etc/sysconfig/network-scripts/ifcfg-eth*" files will create enable interfaces at boot time, without any associated IP address.

```
DEVICE="eth4"

HWADDR="00:1B:21:82:C2:A1"

NM_CONTROLLED="yes"

ONBOOT="yes"

DEVICE="eth5"

HWADDR="00:1B:21:82:C2:A0"

NM_CONTROLLED="yes"

ONBOOT="yes"
```

Once configured, the interfaces can be enabled.

```
# ifup eth4

# ifup eth5
```

Finally, a provided script can be used to verify the configuration.

```
# /usr/libexec/fcoe/dcbcheck.sh eth4

DCB is correctly configured for FCoE
```

```
# /usr/libexec/fcoe/dcbcheck.sh eth5

DCB is correctly configured for FCoE
```

At this point, the "fcoe" service can be enabled and configured to start at boot time.

```
# /etc/init.d/fcoe start

Starting FCoE initiator service: [ OK ]

# chkconfig fcoe on
```

After the couple of seconds required for negotiation, the FCoE interfaces should be seen as enabled from the switches perspective. On the Linux host, we can see that the fcoe service has identified the FCoE network properties, including the correct fabric per interface.

```
eth4.300-fcoe Link encap:Ethernet HWaddr 00:1B:21:82:C2:A1

inet6 addr: fe80::21b:21ff:fe82:c2a1/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:167 errors:0 dropped:0 overruns:0 frame:0

TX packets:177 errors:0 dropped:1 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:15690 (15.3 KiB) TX bytes:13904 (13.5 KiB)

eth5.100-fcoe Link encap:Ethernet HWaddr 00:1B:21:82:C2:A0

inet6 addr: fe80::21b:21ff:fe82:c2a0/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:153 errors:0 dropped:0 overruns:0 frame:0

TX packets:164 errors:0 dropped:1 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:14906 (14.5 KiB) TX bytes:13114 (12.8 KiB)

# fipvlan -a

Fibre Channel Forwarders Discovered

interface | VLAN | FCF MAC

------------------------------------------
```

```
eth5 | 100 | 00:0d:ec:f9:35:80

eth4 | 300 | 00:0d:ec:f9:35:80
```

Linux will create entries under the "/sys/class/scsi_host/" for the new software FCoE initiators, these can be rescanned, which after all fabric zoning has been completed will make the initiators appear as HBA objects on the Storage Center just like any other initiator.

```
# echo "- - -" > /sys/class/scsi_host/host4/scan

# echo "- - -" > /sys/class/scsi_host/host5/scan
```

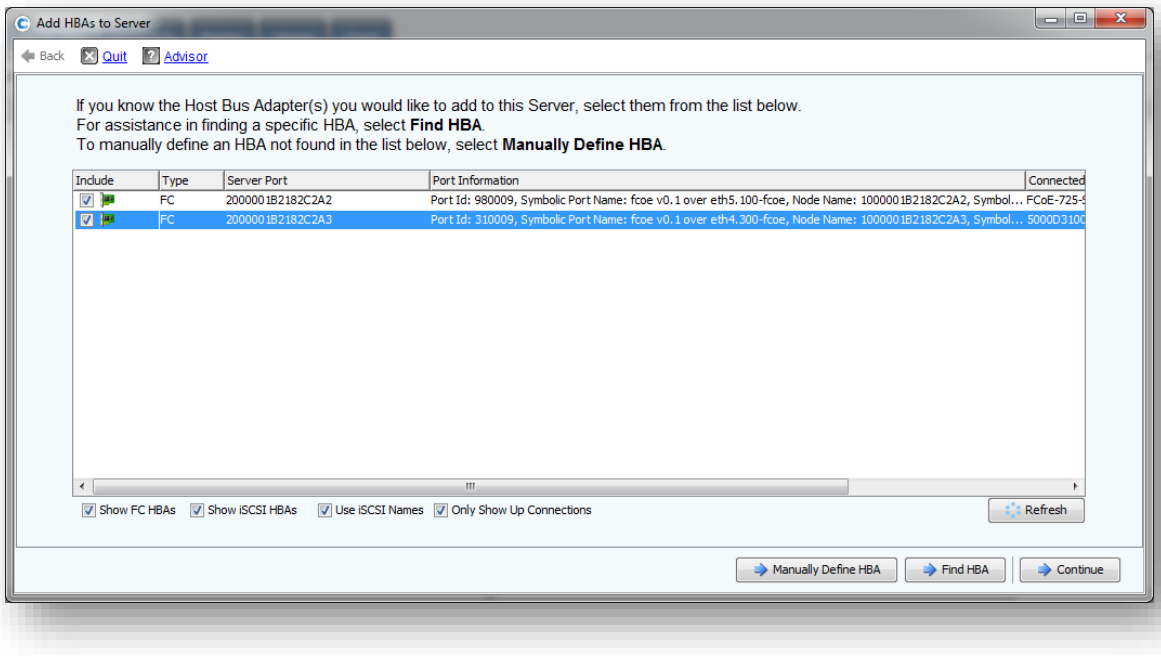
Add HBAs to Server
Back
Quit
Advisor
If you know the Host Bus Adapter(s) you would like to add to this Server, select them from the list below.
For assistance in finding a specific HBA, select Find HBA.
To manually define an HBA not found in the list below, select Manually Define HBA.
Include
Type
Server Port
Port Information
Connected
FC
2000001B2182C2A2
Port Id: 980009, Symbolic Port Name: fcoe v0.1 over eth5.100-fcoe, Node Name: 1000001B2182C2A2, Symbol...
FC
2000001B2182C2A3
Port Id: 310009, Symbolic Port Name: fcoe v0.1 over eth4.300-fcoe, Node Name: 1000001B2182C2A3, Symbol...
Show FC HBAs
Show iSCSI HBAs
Use iSCSI Names
Only Show Up Connections
Refresh
Manually Define HBA
Find HBA
Continue

Figure 8 Correlate Server Port WWPN to FCoE initiator port WWPN

The WWNs can be correlated against both the MAC address of the 10Gb interfaces, and the WWN of the software FCoE initiator as shown above.

```
# ifconfig | grep fcoe

eth4.300-fcoe Link encap:Ethernet HWaddr 00:1B:21:82:C2:A1

eth5.100-fcoe Link encap:Ethernet HWaddr 00:1B:21:82:C2:A0

# cat /sys/class/fc_host/host4/port_name

0x2000001b2182c2a3

# cat /sys/class/fc_host/host5/port_name
```

0x2000001b2182c2a2

2.1 Performance considerations

Despite being a software implementation (as opposed to a hardware implementation i.e. a Converged Network Adapter), the Linux open-FCoE stack provides competitive performance and can be expected to deliver anywhere between 60-80% of the performance as compared to a hardware-based CNA in a similar environment. It is however significantly less "tuneable" to specific environment workloads and because it is a software implementation, it can consume significant amounts of host CPU resources when under load. For this reason, it is suggested that the Linux open-FCoE stack be used only as a secondary tier SAN connectivity solution.

2.2 Special considerations

Because the FCoE services on the host are dependent on the availability and serviceability of the Linux Ethernet stack, some care must be taken with planning and testing of a specific deployment. One suggestion is to use Linux multipath with the "queue_if_no_path" option to manage the volume from the host perspective, even if only one path is available.

Auto-mounting the Linux software FCoE IO stack at boot time is very distribution and environment centric. Please consult the appropriate vendor distribution documentation for further details.

A Additional resources

Dell Compellent Redhat Enterprise Linux (RHEL) 6x Best Practices
http://en.community.dell.com/techcenter/extras/m/white_papers/20437964.aspx